UNIVERSITY OF LINCOLN

A ROBOT IN A BOX:

ADVENTURES IN GPU-ACCELERATED ROS2 DEVCONTAINERS FOR EDUCATION AND ASSESSMENT

Prof Marc Hanheide

Lincoln Centre for Autonomous Systems (L-CAS)

Context: CMP3103 & CMP9767

BSc (Hons) Computer Science with Artificial Intelligence

MSc Robotics and Artificial Intelligence

The Problem - Traditional Setup Hell

"It works on my machine" syndrome

Students spend hours/days on environment setup

Multiple OS dependencies (Windows, macOS, Linux)

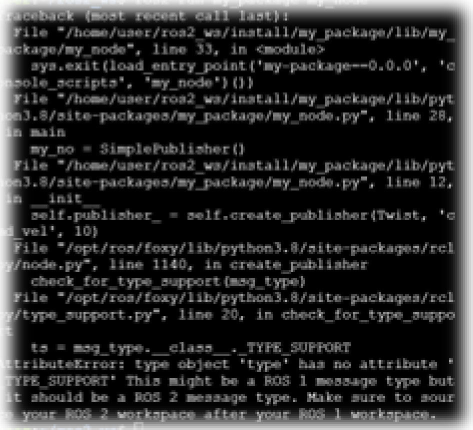
CUDA/GPU driver conflicts

Version mismatches between ROS2, Python(!!), packages, and dependencies

Teaching time lost to troubleshooting

Inconsistent environments across cohort

Result: More time debugging than learning robotics

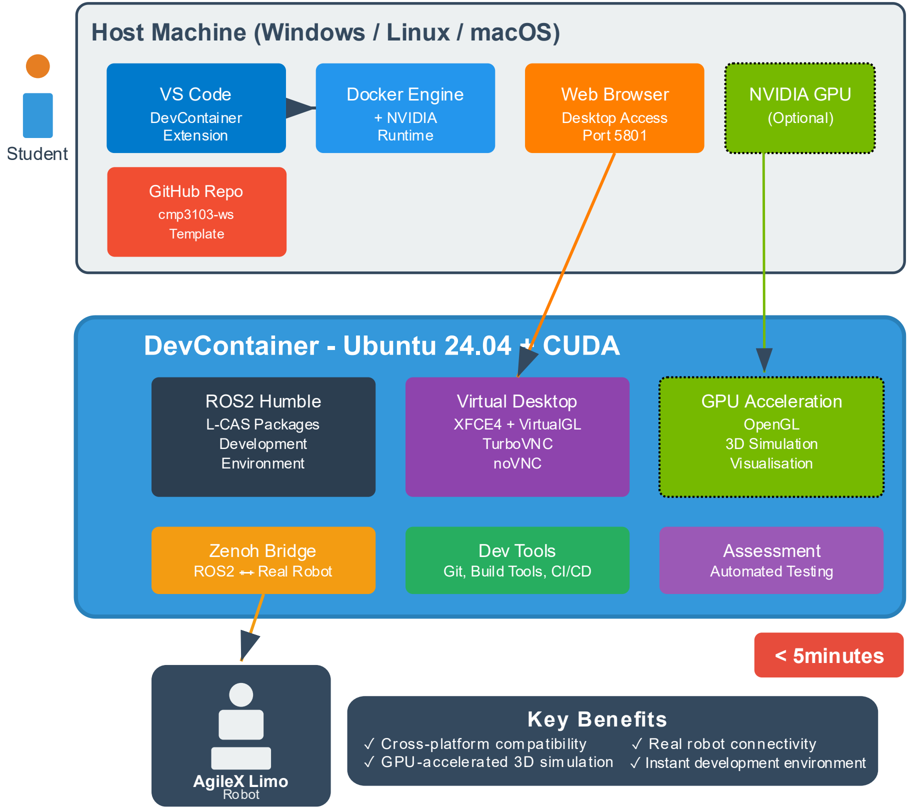


```
aceback (most recent call last):
 File "/home/user/ros2_ws/install/my_package/lib/my_
ckage/my_node", line 33, in <module>
   sys.exit(load_entry_point('my-package==0.0.0', 'c
nsole_scripts', 'my_node')())
 File "/home/user/ros2_ws/install/my_package/lib/pyt
on3.8/site-packages/my_package/my_node.py", line 28,
in main
   my_no = SimplePublisher()
 File "/home/user/ros2_ws/install/my_package/lib/pyt
on3.8/site-packages/my_package/my_node.py", line 12,
in __init__
   self.publisher_ = self.create_publisher(Twist, 'c
d_vel', 10)
 File "/opt/ros/foxy/lib/python3.8/site-packages/rcl
y/node.py", line 1140, in create_publisher
   check_for_type_support(msg_type)
 File "/opt/ros/foxy/lib/python3.8/site-packages/rcl
y/type_support.py", line 20, in check_for_type_suppo
t
   ts = msg_type.__class__._TYPE_SUPPORT
ttributeError: type object 'type' has no attribute '
TYPE_SUPPORT' This might be a ROS 1 message type but
it should be a ROS 2 message type. Make sure to sour
 your ROS 2 workspace after your ROS 1 workspace.
```

Our Solution - Robot in a Box

- Based on GitHub Repository Template: https://github.com/LCAS/ros2_pkg_template/
- DevContainer + Virtual Desktop
- Development Container:
  Fully configured ROS2 environment
- Virtual Desktop: Web-based 3D accelerated interface (VirtualGL + noVNC)
- Cross-platform: Windows, Linux, macOS hosts
- GPU Support: NVIDIA runtime for simulation and visualisation, and CUDA development
- Rapid setup: From clone to working environment
- Isolated container, network no “DDS leakage”
- Zenoh Bridge for talking to robot (running cross compiled version of our docker image on robots)

Robot in a Box: DevContainer Architecture



Student

Host Machine (Windows / Linux / macOS)

VS Code
DevContainer Extension

Docker Engine
+ NVIDIA Runtime

Web Browser
Desktop Access
Port 5801

NVIDIA GPU
(Optional)

GitHub Repo
cmp3103-ws
Template

DevContainer - Ubuntu 24.04 + CUDA

ROS2 Humble
L-CAS Packages
Development Environment

Virtual Desktop
XFCE4 + VirtualGL
TurboVNC
noVNC

GPU Acceleration
OpenGL
3D Simulation
Visualisation

Zenoh Bridge
ROS2 ↔ Real Robot

Dev Tools
Git, Build Tools, CI/CD

Assessment
Automated Testing

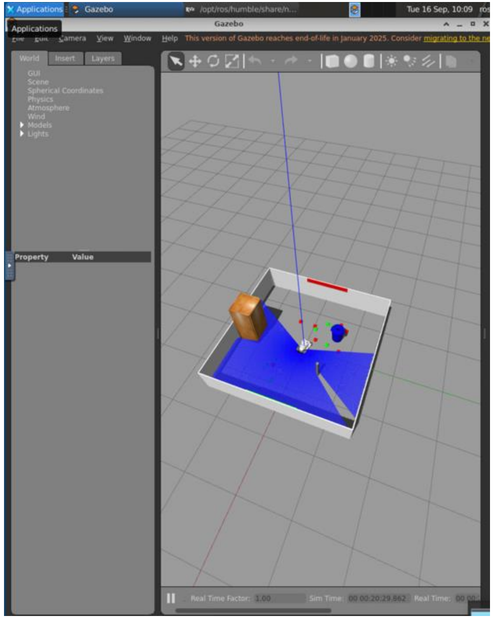
< 5minutes

AgileX Limo
Robot

Key Benefits

✓ Cross-platform compatibility
✓ GPU-accelerated 3D simulation
✓ Real robot connectivity
✓ Instant development environment

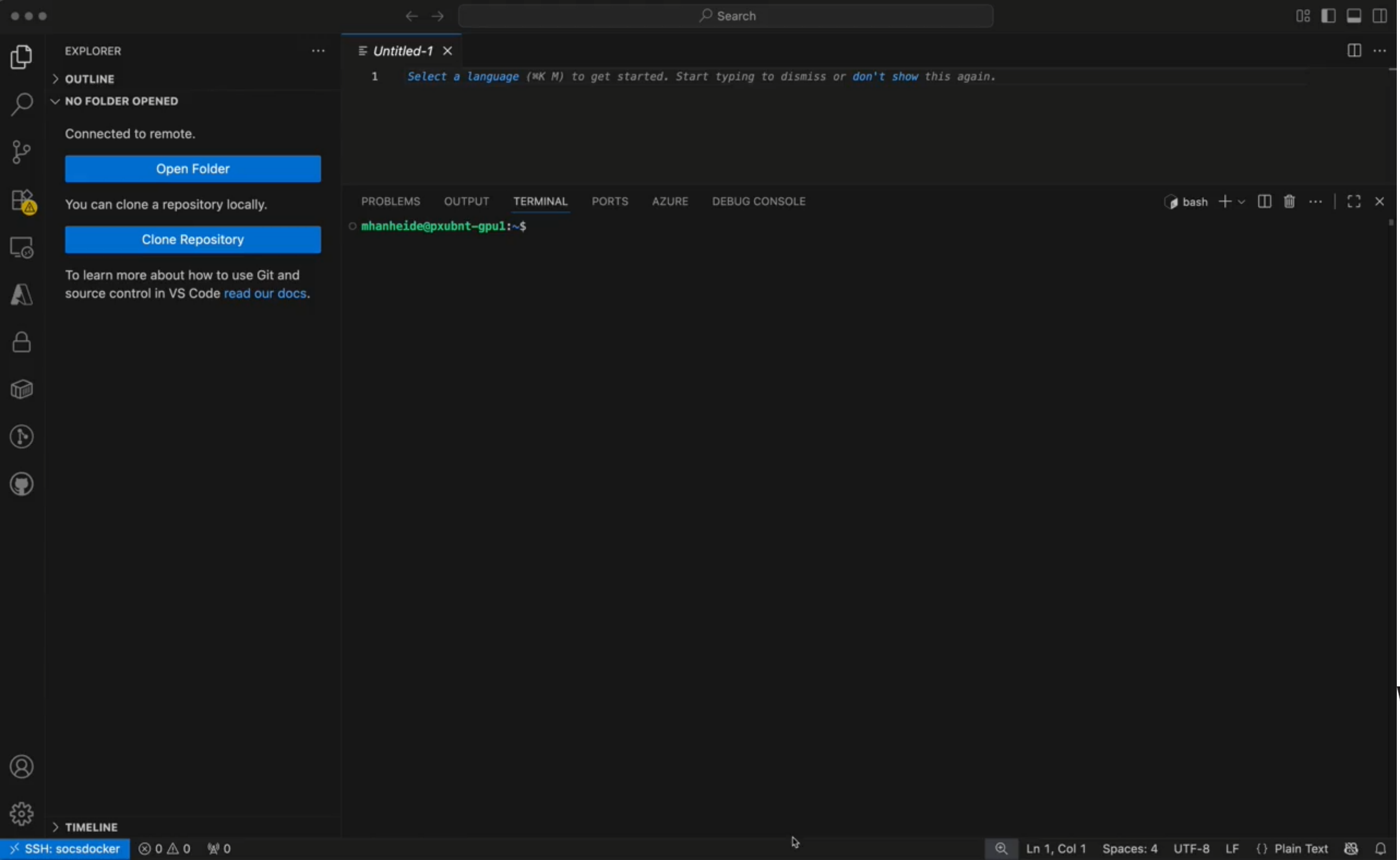
Technical Architecture



This version of Gazebo reaches end-of-life in January 2025. Consider migrating to the ne

Real Time Factor: 1.00 Sim Time: 00 00:20:29.862 Real Time: 00 00:

- Built on Production-Ready Technologies
- Base Image: nvidia/cuda:XX-runtime-ubuntuXX
- ROS Framework: ROS2 Humble (with Cyclone DDS by default)
- Virtual Desktop: VirtualGL + TurboVNC + noVNC
- Development: DevContainer integration
- Communication: Zenoh bridge (built-in container) for real robot connectivity
- Desktop Environment: XFCE4 with GPU acceleration (with NVIDIA)
- Containerised, consistent, and cross-platform



Connected to remote.

Open Folder

You can clone a repository locally.

Clone Repository

To learn more about how to use Git and source control in VS Code read our docs.

```
1  Select a language (⌘K M) to get started. Start typing to dismiss or don't show this again.
```

```
mhanheide@pxubnt-gpu1:~$
```

Educational Impact

Metric	Traditional Setup	DevContainer Solution	Improvement
Setup Time	2-3 workshop sessions	5 minutes	Order of magnitude reduction
Success Rate	~60%	98%	38% increase
Support Tickets	High volume	85% reduction	Major time saving
Teaching Focus	IT troubleshooting	Robotics content	Pure learning
Platform Support	OS-dependent	Universal	Complete coverage

Zenoh Bridge - Connecting Virtual to Real

Seamless Container-to-Robot Communication

- **The Challenge**: DevContainer isolation vs real robot access
- **The Solution**: Zenoh ROS2 bridge enables transparent communication
- Key Benefits:
  - Develop in container, deploy to real hardware with on command line:

    ```
    zenoh-bridge-ros2dds -e tcp/10.82.0.XXX:7447
    ```

  - Students code once, run "everywhere” (and don’t mess with robot installation)

Lessons Learned - Deployment Insights

What Worked ✅

- Web-based desktop removes installation barriers
- GPU passthrough enables proper simulation
- Version control integration maintains consistency
- Cross-platform compatibility works reliably
- Student satisfaction dramatically improved
- Teaching efficiency significantly increased

Challenges ⚠️

- Network bandwidth for initial pulls with very many students
- Windows Docker Desktop permissions
- Container resource management on shared machines
- Initial learning curve for DevContainer concept
- Image size and storage requirements

Resources and Links

- Try yourself (e.g. in code spaces): https://github.com/UoL-SoCS/cmp3103-ws/tree/roscon25
- Useful ROS2 workspace repository template: https://github.com/LCAS/ros2_pkg_template/
- Our instructions for students to use it: https://github.com/LCAS/teaching/wiki/CMP3103
- The Lincoln Centre for Autonomous Systems (L-CAS): https://lcas.lincoln.ac.uk/

Adopted by Tom Howard at Sheffield Robotics